RECEIVED
CENTRAL FAX CENTER
JUL 19 2006

Doc Code: AP.PRE.REQ

PTO/SB/33 (07-05)
Approved for use through xx/xx/200x. OMB 0651-00xx
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

PRE-APPEAL BRIEF REQUEST FOR REVIEW

Docket Number (Optional): RA-5623

I hereby certify that this correspondence is being faxed to the USPTO central fax number (571) 273-8300 (37 CFR 1.8(a))

on July 19, 2006

Signature: Beth L McMahon

Typed or printed name: Beth L. McMahon

Application Number	Filed
10/620,406	7/16/2003

First Named Inventor: Kelvin S. Vartti

Art Unit	Examiner
2186	Lev Iwashko

Applicant requests review of the final rejection in the above-identified application. No amendments are being filed with this request.

This request is being filed with a notice of appeal.

The review is requested for the reason(s) stated on the attached sheet(s).
Note: No more than five (5) pages may be provided.

I am the

- ☐ applicant/inventor.
- ☐ assignee of record of the entire interest. See 37 CFR 3.71. Statement under 37 CFR 3.73(b) is enclosed. (Form PTO/SB/96)
- ☒ attorney or agent of record. Registration number 41,987
- ☐ attorney or agent acting under 37 CFR 1.34. Registration number if acting under 37 CFR 1.34 _____

Beth L McMahon
Signature

Beth L. McMahon
Typed or printed name

(651) 635-7893
Telephone number

July 19, 2006
Date

NOTE: Signatures of all the inventors or assignees of record of the entire interest or their representative(s) are required. Submit multiple forms if more than one signature is required, see below*.

☒ *Total of 3 forms are submitted. Notice of Appeal, Pre-Appeal Brief Request, Reasons for Request

RECEIVED
CENTRAL FAX CENTER
JUL 19 2006

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
Group Art Unit: 2186
Examiner: Lev Iwashko

Customer Assg. No.	027516	**July 19, 2007**
Serial No.:	10/620,406	
Filed:	7/16/2003	
In re Application of:	Vartti et al.	
Title:	Programmable Cache Management System and Method	
Docket No.:	RA-5623	

Mail Stop AF
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Pre-Appeal Brief Request for Review – Reasons for Request

Dear Sir,

This is in response to the Examiner's Office Action that was mailed ***4/25/2006*** ("Final Rejection") that finally rejects all pending Claims 1-37.

Before considering the rejection in detail, Applicants' invention, which is a mechanism for managing a shared memory system, is summarized for discussion purposes. In one embodiment of the invention, data obtained from a main memory may be retained within a processor node that includes one or more instruction processors (IPs) and one or more caches. (Applicants' Figure 1.) For any given data item retained by the processor node, cache tag logic associated with that processor node may, but need not, record the presence of that data item. Whether the cache tag logic records the presence of a particular data item depends on the state of one or more programmable indicators, as well as attributes associated with the data item. Examples of such attributes include the type of request that caused the data to be provided to the processor node (e.g., read vs. write request), the identity of the entity that requested the data, and the type of access rights provided by the main memory along with the data. For instance, the programmable indicators may be set to a state that determines

CERTIFICATE OF MAILING (37 CFR 1.8(a))

I hereby certify that this correspondence is being transmitted via facsimile to the Patent and Trademark Office at (571) 273-8300 on the date shown.

Beth McMahon — July 19, 2006
(Beth McMahon)

that only data provided in a read-only state to any of a predetermined sub-set of IPs will be tracked by the cache tag logic.

When Applicants' cache tag logic is updated to track the presence of a data item within a node according to the above-described mechanism, that data item may, but need not, be stored to a cache memory that is associated with the cache tag logic. The state of the indicators determines whether this replacement operation is performed for a particular data item.

Applicants' independent Claims were finally rejected under 35 USC §102(b) as anticipated by U.S. Pat. No. 5,913,224 to MacDonald ("MacDonald"). MacDonald discloses a system wherein a processor uses a transaction look-aside buffer (TLB) to determine whether an upcoming read request involves retrieval of real-time code. (MacDonald Fig. 2.) If so, the processor activates its lock signal 161, which is provided to a cache management unit (CMU) 202 during the read request. If the request results in a cache miss, the real-time code is retrieved from system memory 300 and is stored in a cache line that is locked by setting a respective lock bit. This prevents the real-time code from being aged from the cache until execution of the code completes. When data/code that is not "real-time" is fetched, the processor's lock signal is not activated, and data fetched from main memory is stored in cache without setting a respective lock bit. This data may then be aged from cache using least-recently used (LRU) algorithms.

Next, Applicants' Claim 1 is considered. This Claim includes:

a programmable storage device to store one or more indicators;
a cache;
cache tag logic; and
a control circuit ... to receive data for possible retention in the cache and to determine, based on the state of the one or more indicators, whether to update the cache tag logic to track the data.

The Examiner asserts that the MacDonald CMU teaches Applicants' control circuit. The Examiner further asserts that a MacDonald memory controller that is contained within the CMU teaches Applicants' indicators. (Final

Rejection p. 3 lines 1-5.) This rejection is improper for at least the following reasons:

1. In MacDonald, it appears that any time the CMU receives data from system memory following a cache miss, the data is stored in cache and the cache tags are updated. For instance, when the CMU receives data that is not real-time code, that data is processed "normally" by storing the data in cache, updating the tags, and leaving the lock bits in a deactivated state so that the data is eligible for replacement according to a LRU algorithm. (MacDonald col. 6 lines 9-14 and col. 10 lines 53-55.) In a similar manner, when the CMU receives real-time code from system memory, that code is also stored in the cache and the tags are updated. In this latter case, however, the CMU locks the real-time code in cache by setting the lock bits for the cache lines storing that code.

It is important to note that nothing in MacDonald states that retrieved code/data that is not "real-time" will not be cached. In fact, MacDonald specifically describes that code/data that is not "real-time" is replaced in cache in the following manner in the embodiment of the MacDonald shown in Fig. 5:

> "Comparator 285 indicates to replacement logic 287 on lines 288 that the **address is not a real-time code address**. In response, replacement logic 287 directs the memory interface to retrieve the data corresponding to the address from system memory 300 and **store it in data way 0 preferably according to the least recently used algorithm** described previously." (MacDonald col. 10 lines 53-55 describing the embodiment of Fig. 5, emphasis added.)

Moreover, if the MacDonald system only performed cache replacement operations for the real-time code, the cache lock bits 230 of Fig. 2 would not be necessary, since *any* cache line storing valid data could be considered locked.

2. The Examiner states that the MacDonald memory controller teaches Applicants' indicators which are stored in a programmable storage device. (See Final Rejection page 3 lines 1-5.) The only reference to the MacDonald memory controller, which does not appear in the drawings, is as follows:

> "Cache management unit 202 preferably includes a memory controller for providing access to L2 cache memory 201. The memory controller may be any one of a number of commonly known memory controllers compatible with the

selected CPU core 110 and overall computer architecture." (MacDonald col. 5 lines 6-11.)

Beyond this brief reference, MacDonald is silent about how that memory controller operates. There is no teaching in MacDonald that describes this memory controller as providing any programmable storage device for storing indicators. Moreover, there is nothing in MacDonald that describes the CMU (cited as teaching Applicants' control circuit) as using any stored state of the memory controller to determine whether to update cache tag logic to track received data, as claimed by Applicants' Claim 1.

3. Although the Examiner does not raise this point, the method of MacDonald Fig. 4 is considered for completeness. Step 425 of Fig. 4 relates to the retrieval of code that is not "real-time". MacDonald states that when this occurs, that code is fetched and executed according to known protocols. (See MacDonald col. 8 lines 18-20.) Throughout the MacDonald description, "known protocols" of fetching the data are described as the processor making a memory request without the lock output signal 161 activated so that the code or data that is not real-time may be stored in the L2 cache using known LRU algorithms and without setting the lock bit. As discussed above, nothing in MacDonald implies that if a TLB lock bit is not activated, the associated data will not be cached. Stated otherwise, it will be appreciated from a complete reading of MacDonald that MacDonald Fig. 4 does not stand for the proposition that only "real-time" code is stored in cache.

To summarize, the rejection of Claim 1 is improper as follows:

1. Nothing in MacDonald, including the CMU, teaches a control circuit that receives data for possible retention in cache, and which then determines whether to update the cache tag logic to track this data based on indicators stored in a programmable device; and

2. Nothing in MacDonald, including the memory controller (as cited by the Examiner), the TLB lock bits, the processor's lock output signal, or the cache lock bits, teaches Applicants' indicators stored within a programmable storage device that are used by the control circuit in the above-described manner.

For at least the foregoing reasons, a prima facie case of rejection has not been made and the rejection of Claim 1 is improper. The remaining independent Claims include aspects similar to Claim 1 and are allowable for reasons similar to those discussed above. Likewise, each dependent Claim depends from a respective independent Claim and is allowable for at least the reasons discussed in reference to these Claims. These Claims include additional aspects not taught or suggested by MacDonald. Attention is directed to the response of 03/01/06 for a detailed discussion of the dependent Claims as they relate to MacDonald.

Finally, dependent Claims 15 and 27 are considered in more detail. These Claims, which were rejected under 35 USC §103(a) based on MacDonald (Final Rejection p. 11), describe Applicants' mode switch logic. This logic is provided to automatically re-program the indicators in response to monitored system conditions (e.g., cache miss rate) so that performance is optimized. The Examiner cites the MacDonald write-back circuit as suggesting this aspect of the invention. (Final Rejection p. 11.) This write-back circuit, which is not shown in the MacDonald drawings or described in any detail, is said to control the storing of updated data from the cache to system memory. (MacDonald col. 5 lines 51-52.) Recall that the Examiner cites the MacDonald memory controller as teaching Applicants' indicators. (Final Rejection page 3, lines 1-5.) Following this reasoning, the Examiner appears to be stating that the MacDonald write-back circuit re-programs indicators provided by the memory controller. However, nothing in MacDonald describes this type of operation. Moreover, nothing describes the write-back circuit as re-programming *any* indicators at all, or that any such re-programming is based on monitored conditions. For at least these additional reasons, the rejection of Claims 15 and 27 is improper. In view of the foregoing, relief is requested from the Final Rejection of all the Claims.

Respectfully submitted,
Beth L. McMahon

Beth L. McMahon
Attorney for Applicants, Reg. No. 41,987
Telephone No. (651) 635-7893
Unisys Corporation, M.S. 4773,
P.O. Box 64942, St. Paul, MN 55164-0942